

Amendments to the Claims:

Rewrite the claims as set forth below. This listing of claims replaces all prior versions and listings of claims in the application:

Listing of the Claims:

1. (Previously Presented) A method for hierarchical Z buffering and stenciling comprising:

comparing a tile Z value range of a tile with a hierarchical Z value range and a stencil code;

updating the hierarchical Z value range and the stencil code in response thereto; and determining whether to render a plurality of pixels within the tile based on the comparison of the tile Z value range with the hierarchical Z value range and the stencil code.

2. (Original) The method of claim 1 wherein the step of determining whether to render a plurality of pixels further comprises:

determining at least one of the following: (1) if a stencil test fails, (2) if the stencil test passes and a hierarchical Z value test fails and (3) if the stencil test passes and the hierarchical Z value test passes on at least one pixel in the tile.

3. (Original) The method of claim 2 further comprising:

rendering the pixels of the tile when the stencil test passes and the hierarchical Z value test passes on at least one pixel in the tile.

4. (Original) The method of claim 2 further comprising:

killing the tile when at least one of the following occurs: the stencil test fails or the stencil test passes and the hierarchical Z value test fails.

5. (Original) The method of claim 1 wherein the tile Z value range contains a tile MinZ and a tile MaxZ and the hierarchical Z value range contains a hierarchical cache MinZ and a hierarchical cache MaxZ.

6. (Original) The method of claim 5 wherein the stencil code is a three bit data value representing a range of stencil values in the tile relative to a background value.

7. (Original) The method of claim 1 wherein the step of determining whether to render a plurality of pixels further comprises:

determining if a per-pixel depth operation needs to be performed; and
determining if stencil operations need to be performed.

8. (Previously presented) A method for hierarchical Z buffering and stenciling comprising:

receiving a tile having a plurality of pixels;

determining if the tile is visible relative to a stencil;

determining if the tile is visible in a hierarchical Z plane; and

updating a hierarchical Z value range and a stencil code in response thereto, wherein the stencil code is a multiple-bit indicator which specifies a relation of a plurality of stencil values in the tile relative to a background value.

9. (Previously Presented) The method of claim 8 further comprising:
generating an indicator to indicate whether to render the plurality of pixels within the tile.

10. (Currently Amended) The method of claim 9 wherein generating the indicator the
~~decision whether to render the plurality of pixels within the tile~~ includes:
determining if a per-pixel depth operation needs to be performed; and
determining if stencil operations need to be performed.

11. (Currently Amended) The method of claim 8 wherein determining if the tile is
visible ~~in a hierarchical Z plane relative to the stencil~~ comprises:
generating [[a]]the stencil code; and
comparing the stencil code to a stencil value and a stencil mask.

12. (Cancelled)

13. (Currently Amended) The method of claim 11 wherein determining if the tile is
visible in a hierarchical Z plane comprises:
receiving a MinZ and a MaxZ for the tile; and
comparing the MinZ and the MaxZ to a hierarchical Z ~~range; and range,~~ wherein at least
one of the plurality of pixels is visible in the z-plane, indicating the tile is visible in the
hierarchical Z plane.

14. (Currently Amended) The method of claim ~~9-12~~ comprising generating an indicator, wherein the indicator indicates ~~at least one of the following~~: a positive indication when it is determined that the tile is visible relative to the stencil and it is determined that the tile is visible in the hierarchical Z plane and wherein the indicator indicates a negative indication when it is determined that the tile is not visible relative to the stencil or it is determined that the tile is not visible in the hierarchical Z plane.

15. (Currently Amended) The method of claim 14 wherein the pixels of the tile are rendered if the indicator indicates [[a]] the positive indication and wherein the tile is killed if the indicator indicates [[a]] the negative indication.

16.-18. (Cancelled)

19. (Currently Amended) The apparatus of claim 23 wherein the comparator compares the tile MinZ and the tile MaxZ to the cache MinZ and the cache MaxZ, and compares the stencil code to a stencil value and a stencil mask to determine determines if a per-pixel depth operation needs to be performed and to determine determines if stencil operations need to be performed.

20.-22. (Cancelled)

23. (Currently Amended) A graphics processing engine comprising:

a comparator operative to receive a tile MinZ and a tile MaxZ associated with a tile having a location and a plurality of pixels;

a hierarchical Z buffer and stencil cache operably coupled to the comparator; and

a hierarchical Z buffer and stencil cache updater operably coupled to the ~~comparator~~ comparator, wherein the hierarchical Z buffer and stencil cache provides a cache MinZ, cache MaxZ, and a stencil code to the comparator, wherein the cache MinZ and the cache MaxZ are associated with the location of the tile and wherein the comparator compares the tile MinZ and the tile MaxZ to the cache MinZ and the cache MaxZ, and compares the stencil code to a stencil value and a stencil mask.

24. (Previously Presented) The apparatus of claim 23 wherein the comparator generates an indicator that indicates the visibility of the plurality of pixels of the tile relative the stencil mask and a hierarchical Z plane.

25. (Previously Presented) The apparatus of claim 24 further comprising:

a kill module operably coupled to the hierarchical Z buffer and stencil cache updater wherein the hierarchical Z buffer and stencil cache updater receives the indicator from the comparator and the hierarchical Z buffer and stencil cache updater provides a kill signal to the kill module based on the indicator and wherein the hierarchical Z buffer and stencil cache updater updates the hierarchical Z buffer and stencil cache in response to the indicator.

26. (Previously presented) The method of claim 1, wherein comparing the tile Z value range with the hierarchical Z value range and the stencil code comprises comparing the stencil code to a stencil value and a stencil mask.

27. (Previously Presented) A method for hierarchical Z buffering and stenciling comprising:

determining if a tile is visible relative to a stencil by generating a stencil code and comparing the stencil code to a stencil value and a mask; and

determining if the tile is visible in a hierarchical Z plane by comparing a MinZ and a MaxZ for the tile to a hierarchical Z range.

28. (Currently Amended) The method of claim 27, further comprising:

generating an indicator wherein the indicator indicates a positive indication when the tile is visible relative to the stencil and visible in the hierarchical Z plane, and wherein the indicator indicates a negative indication when the tile is not visible relative to the stencil or in the hierarchical Z plane; [[and]]

rendering pixels of the tile if the indicator indicates the positive indication; and

killing the tile if the indicator indicates the negative indication.

29. (Previously Presented) The method of claim 28, further comprising updating the hierarchical Z value range and the stencil code.

30. (Previously Presented) A method for hierarchical Z buffering and stenciling comprising:

determining if a tile is visible relative to a stencil by generating a stencil code and comparing the stencil code to a stencil value and a mask;

determining if the tile is visible in a hierarchical Z plane by comparing a MinZ and a MaxZ for the tile to a hierarchical Z range; and

generating a signal indicating that a detailed depth test is not required because all pixels of the tile are known to be visible in the hierarchical Z plane.